USGS-OFR-85-233-A

UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

POLYGON - An Interactive Program For Constructing and Editing The Geometries of Polygons Using A Color Graphics Terminal

by

Bruce A. Chuchel
U.S. Geological Survey
Menlo Park, CA, 94025

Open File Report 85-233-A

Prepared in cooperation with the
Nevada Operations Office
U. S. Department of Energy
(Interagency Agreement DE-AI08-78ET44802)

This report is preliminary and has not been reviewed for conformity with U.S. Geological Survey editorial standards. Use of brand names in this report is for the sake of description only, and does not constitute endorsement by the U. S. Geological Survey. Although this program has been tested, the U.S. Geological Survey makes no guarantee of correct results.

Menlo Park, California

1985

USGS-OFR-85-233-A

UNITED STATES
DEPARTMENT of INTERIOR
GEOLOGICAL SURVEY

POLYGON - An Interactive Program For Constructing and Editing The Geometries of Polygons Using A Color Graphics Terminal

BY

B. A. CHUCHEL[1]

[1]U.S. Geological Survey, Menlo Park, CA

ABSTRACT

This report is a user manual for the FORTRAN program POLYGON. POLYGON is designed to facilitate the construction and editing of the geometries of polygonal shapes using an Envision color-graphics terminal. The Open-File Report #85-233-B [Chuchel, 1985] contains a listing of the FORTRAN source code for the POLYGON program.

POLYGON features color-graphics, an interactive user dialogue, brief help messages, and "zooming" to selected portions of the terminal screen in order to edit and manipulate polygons using a graphics cursor or the Envision mouse. A quadruply linked tree is used within POLYGON to represent and store the information about the geometries of the polygons.

These polygonal shapes may subsequently be used by independent modeling programs to represent three-dimensional shapes of gravitational or magnetic sources; for example, output from POLYGON is directly compatible with programs MAGPOLY or GRAVPOLY [Plouff, 1975a, 1975b; Godson 1983a, 1983b] and PFGRAV3D or PFMAG3D [Blakely, 1981].

POLYGON is written in DEC (Digital Equipment Corporation) extended FORTRAN 77 and is presently operational on the USGS, Branch of Geophysics, VAX/VMS computer. The program is written to operate on the Envision 200 series of color graphics terminals by Envision Technology Incorporated. Suggestions are provided on converting the program to other types of color terminals.

CONTENTS

Abstract..... 2
CONTENTS..... 3
List of Figures..... 3
Introduction..... 4
Input/Output Files..... 7
Applications to Gravity and Magnetic Modeling..... 8
I. General Instructions..... 11
A. Default answers, obtaining help, quitting a sublevel..... 11
B. Envision mouse button and cursor key assignments..... 11
II. Examples of program operations..... 12
A. Getting started..... 12
B. POLYGON command level..... 14
C. Add/change_parm/delete/edit_poly mode..... 15
C.1 - Adding a polygon..... 17
C.2 - Changing polygon parameters..... 18
C.2.1 - Changing labels..... 18
C.2.2 - Changing parameters of a polygon..... 19
C.3 - Deleting a polygon..... 21
C.4 - Editing a polygon..... 22
C.4.1 - Adding points..... 22
C.4.2 - Deleting a point..... 22
C.4.3 - Moving a point..... 22
C.5 - Special functions..... 23
C.5.1 - Copying a polygon..... 23
C.5.2 - Moving a polygon..... 23
C.5.3 - Rotating a polygon..... 23
D. Output Plouff/Standard model files..... 25
D.1 - Plouff/Godson model..... 25
D.2 - Standard grid..... 27
E. Reading in a POLYGON model..... 28
F. Writing out a POLYGON model..... 28
G. Zoom mode..... 29
H. Quitting the program..... 29
Notes About Implementation..... 30
References..... 32
Appendix A - POLYGON ASCII model format..... 33
B - Standard grid format..... 35
C - POLYGON data structure..... 36

LIST OF FIGURES

Figure 1 - POLYGON Control Structure..... 5
Figure 2 - Example of polygons in plan-view..... 10
Figure 3 - Several possible cross-sections through Figure 2 along line ZZ'..... 10
Figure 4 - Color plot of grid on screen with several polygons..... 16
Figure 5 - Quadruply linked tree data structure for model in Fig. 6.. 37
Figure 6 - A model having the data structure depicted in Fig. 5..... 38

INTRODUCTION

POLYGON is a computer program for developing polygonal models on an Envision color-graphics terminal. The term "polygonal model" shall refer to (1) the coordinates of the corners of a set of polygonal shapes, (2) a set of parameters and parameter descriptions attached to each polygon, and (3) the topologic structure that links the polygons in a storage hierarchy.

The program features color-graphics, an interactive user dialogue, brief help messages, several types of output files, and "zooming" to selected portions of the graphics screen in order to edit and manipulate polygons using a graphics cursor or the Envision mouse. Figure 1 shows the program control structure.

A model may consist of up to 100 polygons; each with up to 100 vertices. Each polygon may be assigned up to ten numerical parameters, each with a descriptive label. A polygon's sides may not be self-crossing; perimeters of polygons are not allowed to cross, but any number of polygons can be completely contained within other polygons.

POLYGON uses a data structure called a quadruply linked tree [Knuth, 1969, p. 352] to store the topology of the collection of polygons composing the model. The data structure is described in Appendix C.

A typical modeling session might include the following steps:

1) The POLYGON program is invoked.
2) A two-dimensional array (grid) is input, stored in program memory, and contoured in color on the graphics screen.
3) A mouse or other cursor control device is used to draw, edit, and manipulate an initial set of polygons on the screen.
4) Parameters are assigned to individual polygons and labels are assigned to each parameter.
5) The screen coordinates of the polygons are converted to the coordinate system of the grid and, together with the parameter information, are written to output files for use in other programs.

Depending on the application, a POLYGON model may require revision. For example, output from POLYGON may be used as initial input to other programs that calculate magnetic anomalies over prismatic sources. Comparison of calculated anomalies with observed anomalies usually indicates ways in which the prismatic sources (i.e., the POLYGON model) can be adjusted in order to improve agreement. The steps to edit an existing POLYGON model are as follows:

6) The POLYGON program is invoked again.
7) A color-contour representation of a grid is displayed in color on the screen as in step 2.
8) The mouse or cursor keys are again used to add and/or modify the polygonal bodies as necessary to agree with some desired criteria.

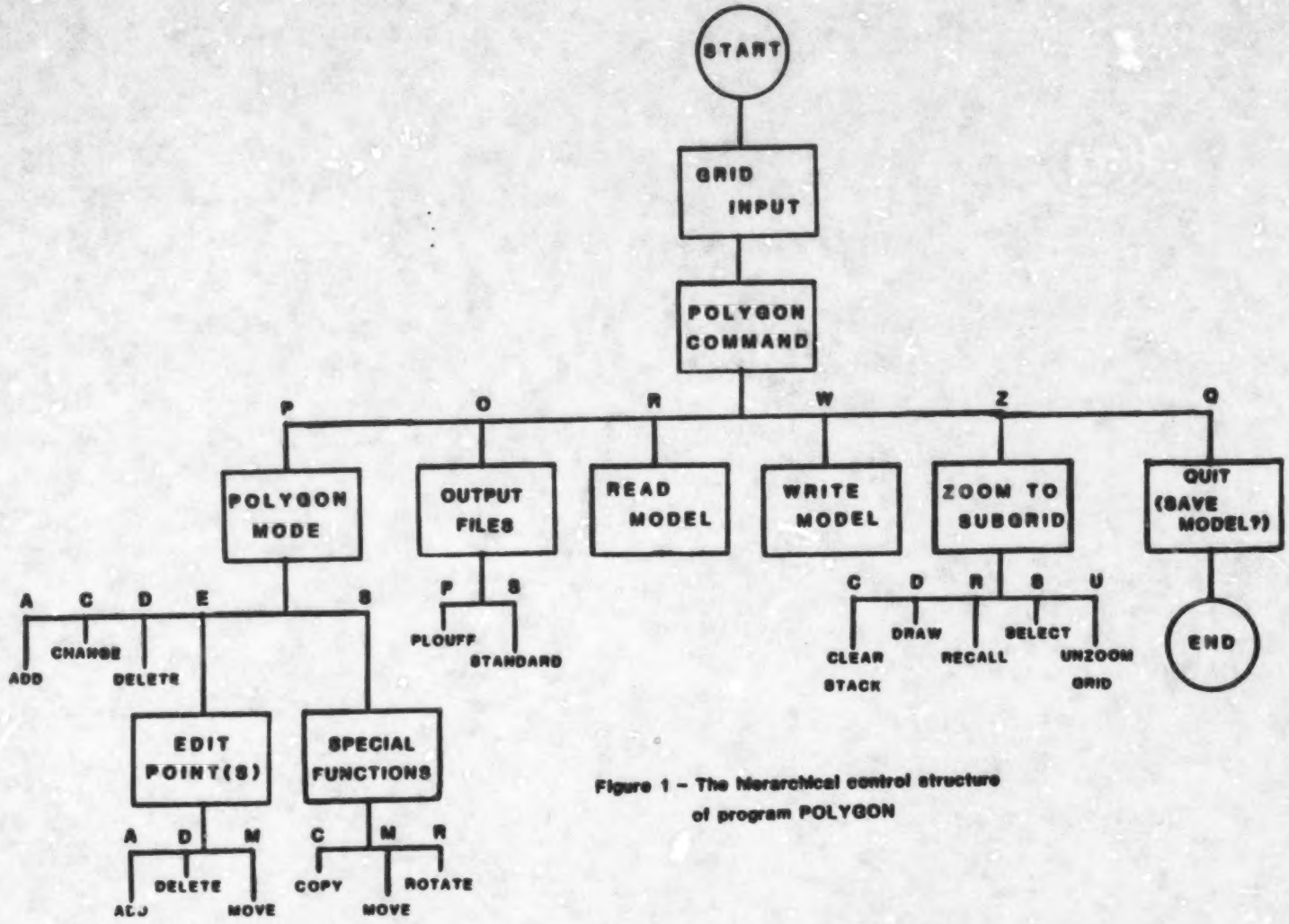


Figure 1 – The hierarchical control structure of program POLYGON

POLYGON is written to operate on the Envision 200 series of color graphics terminals. The 200 series features a Tektronix 4014 compatible instruction set [Envision, 1983b]. Modular construction of the program should make it relatively simple to convert to other color graphics terminals. The section "Notes on implementation" provides some information on converting to other terminals and other versions of FORTRAN.

POLYGON has been developed in support of the U.S. Geological Survey's effort to characterize potential radioactive waste storage sites at the Nevada Test Site for the Nevada Nuclear Waste Storage Investigations project. The Open-File Report of Chuchel [1985b] contains a listing of the FORTRAN source code for the POLYGON program.

INPUT/OUTPUT FILES

POLYGON can be used to generate three types of output files. One of these files, termed a POLYGON model file, may be used in subsequent POLYGON modeling sessions. The other two types of files may be used as input to other programs. The three types of files are described briefly below.

1) POLYGON model file
   This is a formatted ASCII file for use in the POLYGON program. Appendix A describes the format of information contained in this file.
2) Plouff/Godson model
   This file is compatible with programs GRAVPOLY or MAGPOLY which calculate the anomalies caused by a three-dimensional gravity or magnetic source. The formats for these files are described in [Plouff, 1975a, 1975b] and [Godson, 1983a, 1983b].
3) Standard grid file
   A standard grid file is a binary file containing two-dimensional array data. This type of file is fundamentally different from the other two and represents a digital "map" of one parameter of the entire set of polygons. Using this file type, POLYGON is useful in constructing gridded maps of the horizontal variation of some geophysical parameter; e.g., density or magnetic susceptibility. This binary format is compatible with many programs used by the Branch of Geophysics, USGS. Appendix B discusses the format of information contained in this file.

APPLICATIONS TO GRAVITY AND MAGNETIC MODELING

As discussed above, POLYGON can generate several different types of output files for calculating the anomalies from a gravity or magnetic model. The following procedures describe the construction of a gravity or magnetic model:

(1) The POLYGON program is invoked.
(2) A color map of the gridded observed anomaly data is displayed on the screen.
(3) Either a mouse or other graphics cursor control device is used to draw, edit, and manipulate an initial set of polygons representing the source bodies, in plan-view.
(4) Parameters are assigned to individual polygons; e.g., density, magnetization, height of top, height of bottom.
(5) The screen coordinates are converted to the coordinate system of the grid and, together with the parameter information, are used to generate one of several types of output files.
(6) Results from POLYGON are used by independent programs for calculating anomalies.
(7) The POLYGON program is invoked again.
(8) A grid of calculated anomalies (or a grid of the difference between the calculated and observed anomalies) is displayed in color on the screen as in step 2.
(9) The mouse or cursor keys are again used to add and/or modify the geometry of the polygonal source bodies as necessary to improve the agreement between the calculated and observed anomalies.

Depending on the type of output file (step 5), output from POLYGON will be directly compatible with a variety of magnetic and gravity modeling programs. If the user specifies a standard grid file, for example, polygon corner point and parameter information are used to assign values to each grid intersection, those grid intersections not interior to any polygons are assigned to a user entered default value. The standard grid will have the same grid specifications as the grid displayed on the graphics screen. Instead of creating a standard grid the user has the option of writing the polygon parameters onto a preexisting standard grid, which must have the same grid specifications as the grid on the graphics screen. Standard grids are accepted by programs PFMAG3D and PFGRAV3D [Blakely, 1981] as maps of susceptibility or density. These are combined with topographic information according to the method of Parker [1972].

The user may also specify that output be compatible with GRAVPOLY or MAGPOLY. If this option is used, the corner point and parameter information for a body (polygon) should follow these rules:

i) The corners of a body may be entered in either a clockwise or counter-clockwise mannner when viewed from above. However, the GRAVPOLY or MAGPOLY model files are written with the corner points of all the bodies output in a clockwise manner.
ii) A hole may be removed from a larger body by including a second body entirely within the parent body with a density or magnetization contrast of opposite sign from that of the parent body.

GRAVPOLY and MAGPOLY require the following parameters for each body.

Gravity model:

a) Height of top of body (positive direction is upward).
b) Height of bottom of body (" " " ").
c) Density contrast of body.

Magnetic model:

a) Height of top of body (positive direction is upward).
b) Height of bottom of body (" " " ").
c) Volume magnetic susceptibility in emu times 100,000 (magnetization contrast).
d) Remanent or total volume magnetization in emu times 100,000.
e) The declination of remanent or total magnetization in degrees, measured clockwise from the y axis.
f) Inclination of remanent or total magnetization in degrees measured downward from the horizontal plane.

These parameters are easily entered and changed for each polygon in the "Change polygon parameters mode" (Section II.C.2.2). See documentation for MAGPOLY [Godson, 1983a] and GRAVPOLY [Godson, 1983b] for a detailed discussion of these parameters.

Figure 2 depicts three bodies as they might appear in plan-view on the screen, the arrows indicate the direction of corner entry. By assigning different top and bottom heights to the individual bodies, non-laminar models may be constructed. Figure 3 shows examples, in cross-section, of several possible models obtainable when using different top and bottom heights for the individual bodies.

When calculating the anomaly from a model such as b or c in Figure 3, any contribution to the anomaly from the overlap of body C on body A and B must be removed (cross-hatched regions of figure). POLYGON creates a "hidden body" in the Plouff/Godson model file to remove the additional contribution in the calculation step. A "hidden body" has the same plan-view polygonal shape as body A or B, but with top and bottom heights defined by the intersection of bodies C and A or B. The density or magnetization contrast given to a hidden body is of opposite sign from that of the parent body; any other parameters for this body are those of the parent.

The hidden bodies are generated at the time a Plouff/Godson file is requested and are not stored by the program. Hence, the total number of hidden bodies is not included in the total number of polygons allowed by the program.

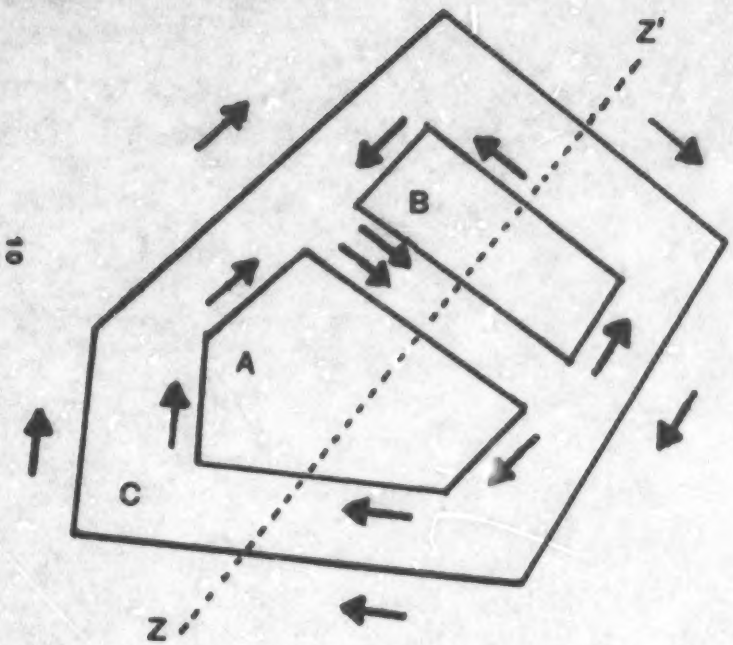


Figure 2 - Polygons in plan-view

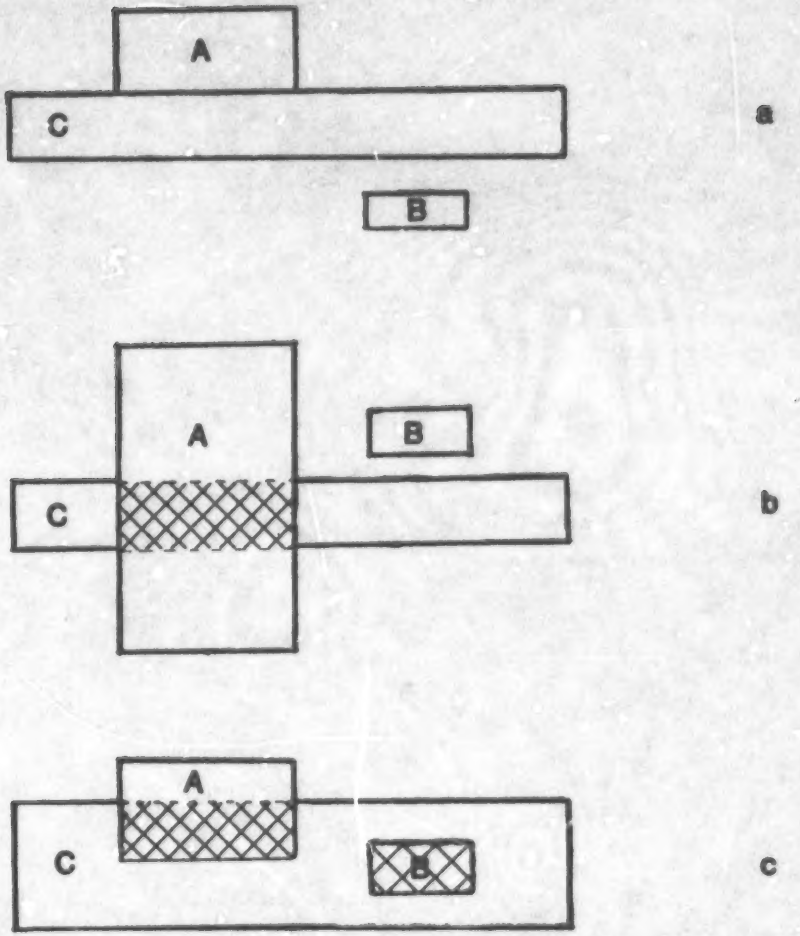


Figure 3 - Several possible cross-sections through Figure 2 along line ZZ'

I. GENERAL INSTRUCTIONS

I.A. DEFAULT ANSWERS, OBTAINING HELP, QUITTING A SUBLEVEL

POLYGON interactively leads the user through model construction. Default answers are available for most of the questions and are indicated in brackets [] at the end of a question. The default answer will be used if the user presses the return key. When specific responses are expected to a question, the choices are indicated in parentheses at the end of the question with different possible responses seperated by a slash (/). For example the question,

"Polygon command (p/o/r/w/z/h/q) [h]?"

indicates that one of the letters "p", "o", "r", "w", "z", "h", or "q" is expected from the user, and that the letter "h" is the default answer.

Responding with the letter "h", for "help", provides a brief message describing the other permissible responses. A user response of "q", "quit", or "//" moves the program from one sublevel to the next higher level.

In the following examples, user responses are underlined. A blank underline indicates that a carriage return <CR> was entered. Answers to questions may be entered in either upper or lower case.

NOTE: Figures in the text showing polygons in black or white are colored in reverse from how they normally appear on the graphics screen.

I.B. ENVISION MOUSE BUTTON AND CURSOR KEYBOARD ASSIGNMENTS

The Envision graphics cursor may be positioned with either the mouse or the cursor control keys on the numeric keypad. POLYGON prompts for which method to use when positioning the graphics cross-hair cursor. The method selected is then used throughout the current invocation of the program. When positioning the graphics cursor the following actions may be performed using the indicated mouse button or keyboard key assignments:

MOUSE BUTTON	KEYBOARD CHARACTER	ACTION
1	"e"	Enters the current cross-hair cursor position as a corner of the polygon
2	"f"	Same as 1 or "e" and finishes entry of the polygon (This only applies when adding polygons or points, otherwise use the same as 1 or "e")
3	"h"	Help message
2&3	"q"	Quit/abort current action

When holding the Envision mouse in the right hand the mouse buttons are numbered from left to right, button one is actuated by the index finger and button three is actuated by the ring finger [Envision, 1983a].

The Envision terminals' graphics cross-hair cursor may be positioned by using the cursor control keys on the numeric keypad; pressing the FUNCT key first causes faster movement of the cursor. The cursor may be moved in both the x and y directions by simultaneously pressing two diagonally-adjacent arrow keys [Envision, 1983b].

II. EXAMPLES OF PROGRAM OPERATIONS
II.A. GETTING STARTED

Upon invoking POLYGON a message is printed containing the version number and creation date of the POLYGON program being executed. POLYGON then prompts:

Question: "Read or create a grid (r/c/q)?"
Response: The user has the option of reading in an existing standard grid (defined in Appendix B), or creating a new one. If the latter is selected, the user will be prompted for the grid specifications. Note that a default answer is not provided. The grid size, the number of rows (NR) times the number of columns (NC), must be less than or equal to 250,000.

a. If "r" for "read" was given as the response, the user is asked the following question. The program dialogue for this information is shown in the example below.
   Q: "Grid name?"
   R: Enter the name of the standard grid to be read in.

b. If "c" for "create" was the response, the grid specifications will be requested. The program dialogue for this information is shown on the next page.

After the grid is successfully read or created, the user is asked the following question to determine the amount of information to be displayed during the current POLYGON session.

Q: "Do you want verbose or terse prompts (v/t/q) [T]?"
R: Enter "v" for verbose, "t" for terse, or "q" to quit.

After this question the user will be placed at the POLYGON command level (fig. 1).

Example:

```
$ RUN POLYGON
*******************************************************************************
     Welcome to POLYGON version 1.0...12/09/84
*******************************************************************************
 Read or create a grid (r/c/q)?r
 Grid name?aero.grd
 Do you want verbose or terse prompts (v/t/q) [T]?_

 POLYGON command (p/o/r/w/z/h/q) [h]?
```

Example:

```
$ RUN POLYGON
*******************************************************************************
   Welcome to POLYGON version 1.0...12/09/84
*******************************************************************************
Read or create a grid (r/c/q)?c

You will now be asked to enter the grid specifications.
(Note: ncol*nrow<=250000),
(Enter // to quit):

    Enter ID for grid (up to 56 characters)? Test of POLYGON, 12/18/84
    NCOL?30
    NROW?60
    xo?0.0
    dx?1.0
    yo?0.0
    dy?1.0
Do you want verbose or terse prompts (v/t/q) [T]?_

POLYGON command (p/o/r/w/z/h/q) [h]?
```

II.B. POLYGON COMMAND LEVEL

Q: "POLYGON command (p/o/r/w/z/h/q) [h]?"
R: Enter the letter of the desired option. Enter "h" or a carriage return to see a list of possible options. Enter "q" or "//" to quit the program; POLYGON asks if the current model should be saved before quitting.

Example:

```
$ RUN POLYGON
*****************************************************************************
   Welcome to POLYGON version 1.0...12/09/84
*****************************************************************************
 Read or create a grid (r/c/q)?r
 Grid name?aero.grd
 Do you want verbose or terse prompts (v/t/q) [T]?_
 POLYGON command (p/o/r/w/z/h/q) [h]?_
 POLYGON options available:
      p = POLYGON (add/change_parm/delete/edit_poly) mode
      o = output Plouff/Godson model or standard grid
      r = read model from file
      w = write model to output file
      z = zoom to subgrid of grid
      q = quit
 POLYGON command (p/o/r/w/z/h/q) [h]?q
 Save model (y/n) [y]?n
 End of POLYGON
$
```

II.C. ADD/CHANGE_PARM/DELETE/EDIT_POLY MODE

Use the add/change_parm/delete/edit_poly mode to enter and make changes to a POLYGON model. The following types of changes may be made:

1) Adding polygons.
2) Changing parameters and labels.
3) Deleting polygons.
4) Editing a polygon by either adding, deleting, or moving corner points.
5) Special functions; copy, move, or rotate a polygon.

Each of the options are described in the following pages. The prompt for the POLYGON add/.../edit_poly mode is:

Q: "POLYGON add/change_parm/delete/edit_poly mode (a/c/d/e/s/h/q) [h]?"
R: Enter a single character of the option desired. Entering a carriage return will display a short explanation of the options available.

The add/.../edit_poly mode is entered by typing "p" when in the POLYGON command level, one of two actions will then occur:

(i) The user will be prompted for parameters to use when coloring the grid on the screen, these parameters will be used throughout the current POLYGON session. After the parameters are entered the grid is drawn in color and the user is placed in the POLYGON add/.../edit_poly mode. The example below shows the dialogue when entering the coloring parameters. Figure 4 is a color plot of the graphics screen after the grid is drawn.
(ii) If the coloring parameters have previously been set the user will be placed immediately in the POLYGON add/.../edit_poly mode.

Example:

```
POLYGON command (p/o/r/w/z/h/q) [h]?P
Here are some facts about your grid:
  MIN, MAX, AVE=  -359.1219          1089.600       -67.05222
  PLUS/MIN 2 SD=  -390.0413          255.9369
To draw the grid in color I first need some parameters:
Enter the number of contours
        0 = Sets up terminal for polygon drawing only
     1-12 = Draws grid with number of contours indicated
       // = Quit
*Number of contours (0-12, // to quit)?12
*Give minimum contour level?-300
*Give contour interval?100
* OK (y/n/q) [Y]?_
```

Figure 4 - Color plot of grid.

II.C.1- ADDING A POLYGON

To add a polygon enter the character "a" for "add" in the POLYGON add/.../edit_poly mode. The corners of a polygon are entered by positioning the graphics cursor to the desired location and then pressing the appropriate key(s); Section I.B describes the key functions. The sides of the polygon are drawn as the corners are entered. The polygon is closed by positioning the graphics cursor to the final corner of the polygon to be enterd and pressing (depending on the method selected) mouse button 2 or the "f" key on the keyboard, the remaining two sides of the polygon are then drawn. Figure 4, on page 16, shows several polygons drawn on the graphics screen.

II.C.2 - CHANGING POLYGON PARAMETERS:

Use the change mode to enter or change the parameters of a polygon, or the labels describing the parameters. The label and parameter information are retained in a POLYGON model file and may be changed at any time. A label may be up to 15 characters in length and the parameters are read using an "e15.8" format.

The change mode is invoked by entering the letter "c" for "change" in the POLYGON add/.../edit_poly mode. The options in the change mode are:

l = Change/assign labels,
p = Change/assign parameters to polygons.

The following example shows the dialogue encountered on entering the change mode.

Example:

```
*** POLYGON add/change_parm/delete/edit_poly mode ***
POLYGON add/change_parm/delete/edit_poly mode (a/c/d/e/s/h/q) [h]?c
*** Change_parm mode ***
Change polygon parameters (l/p/h/q) [h]?_
Change mode options:
       l = Change/assign labels on parameter list
       p = Change/assign parameters to polygons
       q = Quit and return to POLYGON add/.../edit_poly mode
Change polygon parameters (l/p/h/q) [q]?
```

II.C.2.1 - CHANGING LABELS:

Up to ten labels may be entered to describe the parameters associated with the polygons. It is recommended that labels be entered as early as possible in the modeling process. The following diagram shows the relationship between the labels and the parameter fields.

	Label 1	Label 2	... Label 10
Polygon #			
1	parm(1,1)	parm(1,2)	... parm(1,10)
2	parm(2,1)	parm(2,2)	... parm(2,10)
.	.	. .	.
.	.	. .	.
.	.	. .	.
100	parm(100,1)	parm(100,2)	... parm(100,10)

Parameter field

The labels are initialized to the character string "UNASSIGNED". The program dialogue for label entry is shown on the next page.

Example:

```
Change polygon parameters (l/p/h/q) [h]?L
The parameter labels are
   Label  1= UNASSIGNED
   Label  2= UNASSIGNED
   Label  3= UNASSIGNED
   Label  4= UNASSIGNED
   Label  5= UNASSIGNED
   Label  6= UNASSIGNED
   Label  7= UNASSIGNED
   Label  8= UNASSGINED
   Label  9= UNASSIGNED
   Label 10= UNASSIGNED
 Label to change (1-10, -1 for all, 0 or // to quit)?-1
 Enter label number 1 [UNASSIGNED]?Top
 Enter label number 2 [UNASSIGNED]?Bottom
 Enter label number 3 [UNASSIGNED]?Density
 Enter label number 4 [UNASSIGNED]?Vol-MAG
 Enter label number 5 [UNASSIGNED]?Rem-MAG
 Enter label number 6 [UNASSIGNED]?Dec-Mag
 Enter label number 7 [UNASSIGNED]?Inc-Mag
 Enter label number 8 [UNASSIGNED]?Polygon
 Enter label number 9 [UNASSIGNED]?//
 Change polygon parameters (l/p/h/q) [q]?
```

II.C.2.2 - CHANGING PARAMETERS OF A POLYGON

There are two methods for selecting the polygons to view or modify:

1) All polygons; POLYGON "walks" through the tree structure of the polygons, stopping and prompting at each polygon encountered.
2) Picking individual polygons via the graphics cursor.

When walking through the tree structure the user may skip a polygon or the user may quit and return to the "Change parameter mode" section. The following example shows the dialogue occurring when viewing/changing a polygon's parameters. Note that a carriage return enters the default answer and that "//" may be used to stop modifying the polygon being viewed. The parameters are initialized to zero. The dialogue for parameter entry is shown in the following example.

Example:

```
Change polygon parameters (l/p/h/q) [q]?P
*** Change parameters ***
Change parameter mode (a/p/h/q) [H]?_
 Options for changing polygon parameters:
       a = All polygons (via tree structure)
       p = Pick individual polygons using cursor
       q = quit
Change parameter mode (a/p/h/q) [Q]?A
```

(Program dialogue continued on next page.)

```
          (Program dialogue continued from previous page.)
 You will now be asked to assign parameters to be associated with the
    polygon drawn in white.  Enter a carriage return <CR> to use the
    default answer in brackets [].
Enter a carriage return <CR> to continue:_
View/change parameters of enhanced (white) polygon (y/n/q) [Y]?_
Enter // to quit assigning parameters to this polygon

Top             parameter [0.00000000E+00]?3.34
Bottom          parameter [0.00000000E+00]?1.34
Density         parameter [0.00000000E+00]?2.67
Vol-Mag         parameter [0.00000000E+00]?0.0002
Rem-Mag         parameter [0.00000000E+00]?_
Dec-Mag         parameter [0.00000000E+00]?//
```

II.C.3 - DELETING A POLYGON:

To delete a polygon type "d" for "delete" in the POLYGON add/change parm/delete/edit_poly mode.

A polygon is deleted by positioning the graphics cursor to a corner of the polygon and by pressing the appropriate key(s). POLYGON searches for the nearest polygon to the point entered. If no polygon is found an error message is printed and the cursor may be repositioned to find the polygon. When the polygon is located, it is "enhanced" in a different color (usually white), and the user is asked to confirm whether or not this is the correct polygon. The deleted polygon is redrawn in another color (usually black) to verify that the polygon has been removed from the model. The following example illustrates the dialogue occurring during polygon deletion.

Example:

```
POLYGON add/change_parm/delete/edit_poly mode (a/c/d/e/s/h/q) [h]?d
*** Delete polygon mode ***
   *** Enter screen location ***
          (Here the cursor would be positioned over a corner of a
           polygon and a key would be entered.)
 Is the polygon drawn in white the correct one (y/n/q) [Y]?_
POLYGON add/change_parm/delete/edit_poly mode (a/c/d/e/s/h/q) [q]?
```

II.C.4 - EDITING A POLYGON:

The edit mode allows the addition, deletion, or moving of corner points of a polygon. The edit mode is invoked by entering "e" for "edit" in the POLYGON add/.../edit_poly mode. The EDIT mode prompt is:

"Edit polygon mode (a/d/m/h/q) [h]?"

The steps when using the editing functions are briefly outlined below.

II.C.4.1 - ADDING POINTS TO A POLYGON:

1) Identify the polygon by positioning the cross-hair cursor at any corner of the polygon and enter the cursor position by pressing the appropriate key(s).
2) Position the cursor at the appropriate side of the polygon where points will be added and identify this side by pressing the appropriate key(s).
3) Select an end point of this side where the additions will start from, and enter this point by pressing the appropriate key(s).
4) Enter the new corner points by moving the cursor and pressing the appropriate key(s).
5) Enter the final corner point to be added by pressing (depending on the method selected) mouse button 2 or "f" on the keyboard.

II.C.4.2 - DELETING A POINT:

1) Identify the polygon by positioning the cross-hair cursor at any corner of the polygon and enter the cursor position by pressing the appropriate key(s).
2) (Optional) A corner of the polygon different from the corner selected in step 1 may be entered.
3) Respond to the question on deleting the selected corner point.

II.C.4.3 - MOVING A POINT:

1) Identify the polygon by positioning the cross-hair cursor to a corner of the polygon to edit and enter the cursor position by pressing the appropriate key(s).
2) (Optional) A corner of the polygon different from the corner selected in step 1 may be entered.
3) Position the cursor to the new location for the corner point, and enter the new corner position by pressing the appropriate key(s).
   Example:

```
POLYGON add/change_parm/delete/edit_poly mode (a/c/d/e/s/h/q) [h]?e_
    *** Edit polygon mode ***
Edit polygon mode (a/d/m/h/q) [h]?_
Edit mode options:
      a = Add points
      d = Delete a point
      m = Move a point
      q = Quit and return to POLYGON add/.../edit_poly mode
Edit polygon mode (a/d/m/h/q) [h]?
```

II.C.5 - SPECIAL FUNCTIONS:

The special-functions command may be used to copy, move, or rotate a polygon. The special-functions command is invoked by entering "s" for "special" in the POLYGON add/.../edit_poly mode. The special-functions prompt is:

"Special Functions (c/m/r/h/q) [h]?"

When using the special functions a polygon is not allowed to be transformed to a region outside of the unzoomed grid. The dialogue for the special functions is illustrated on the next page. The steps when using the special functions are briefly outlined below.

II.C.5.1 - COPYING A POLYGON:

1) Identify the polygon by positioning the cross-hair cursor at any corner of the polygon to copy, and enter the cursor position by pressing the appropriate key(s).
2) (Optional) A corner of the polygon different from the corner selected in step 1 may be entered.
3) Position the cursor at the location for this corner of the new polygon and enter this position by pressing the appropriate key(s).

II.C.5.2 - MOVING A POLYGON:

1) Identify the polygon by positioning the cross-hair cursor to a corner of the polygon and enter the cursor position by pressing the appropriate key(s).
2) (Optional) A corner of the polygon different from the corner selected in step 1 may be entered.
3) Position the cursor at the new location for the polygon and enter this position by pressing the appropriate key(s).

II.C.5.3 - ROTATING A POLYGON:

1) Identify the polygon by positioning the cross-hair cursor to a corner of the polygon and enter the cursor position by pressing the appropriate key(s).
2) (Optional) A corner of the polygon different from the corner selected in step 1 may be entered.
3) The user is asked if this corner point will be used for the center of rotation.
   a) If not, the cursor may be positioned and a center for rotation may be entered.
4) A rotation angle is entered; the rotation angle is measured in degrees counter-clockwise.

Example:

```
POLYGON add/change_parm/delete/edit_poly mode (a/c/d/e/s/h/q) [h]?s
   *** Special functions mode ***
Special functions mode (c/m/r/h/q) [H]?_
Special functions options:
      c = Copy a polygon
      m = Move a polygon
      r = Rotate a polygon
      q = Quit and return to POLYGON add/.../edit_poly mode
Special functions mode (c/m/r/h/q) [H]?
```

II.D OUTPUT PLOUFF/STANDARD GRID

The types of output files produced by POLYGON were discussed in the section on Input/Output files. To output a model file enter the character "o" for "output" at the POLYGON command level. POLYGON will respond with the following question.

Q: "Output file mode (p/s/h/q) [h]?"
R: Enter the character of the output file type desired. The output file options are displayed by entering "h" or by entering a carriage return.

The dialogue occuring during creation of each of the possible output files is illustrated in the following pages.

Example:

```
POLYGON command (p/c/r/w/z/h/q) [p]?O
*** Output file mode ***
 Output file mode (p/s/h/q) [H]?
 Output file options:
        p = Plouff/Godson format file (GRAVPOLY/MAGPOLY)
        s = Standard grid file
        q = Quit and return to POLYGON command
 Output file mode (p/s/h/q) [Q]?
```

II.D.1 - PLOUFF/GODSON MODEL:

The Open-File reports of [Plouff, 1975a, 1975b] and [Godson, 1983a, 1983b] discuss in great detail the information needed as input to the modeling programs. The example below shows the order of questioning in POLYGON for the case of a magnetic model.

Example:

```
Output file mode (p/s/h/q) [H]?P
Gravity or magnetic model (g/m/h/q) [G]?M
Name of file containing fieldpoint information? TEST.DAT
Type of data in fieldpoint file (enter H for help) [GRIDDED]?_
Height of fieldpoint grid above the same datum that
   is used to reference the body heights [0.00000000E+00]?_
Name of standard grid containing heights of fieldpoints?TEST.GRD
Identifier for printer output?OUT.OUT
Add a constant to the calculated anomalies (y/n/q)?N
Least-squares comparison/read observed values (y/n/h/q)?Y
What value for LSQS (0-7 integer, // to quit) [0]?5
Units in x&y direction (FEET/KILF/MILE/METR/KILM/H/Q) [KILM]?_
Units in z direction (FEET/KILF/MILE/METR/KILM/H/Q) [FEET]?_
Print switch for body information (1/0) [0]?_
```

(Program dialogue continued on next page.)

```
                (Program dialogue continued from previous page.)
You will now enter the numbers of the labels assigned to the following
   parameters:
      (1) Height of Top of body
      (2) Height of Bottom of body
      (3) Volume magnetic susceptibility (emuX100000)
      (4) Remanent or total volume magnetization (emuX100000)
      (5) Declination of remanent or total magnetization in degrees,
          measured positive clockwise from the direction of the y-axis
      (6) Inclination of remanent or total magnetization in degrees,
          measured positive downward from the horizontal plane
Enter a carriage return <CR> to continue:_
The parameter labels are:
   Label  1: Top
   Label  2: Bottom
   Label  3: Density
   Label  4: Vol-Mag
   Label  5: Rem-Mag
   Label  6: Dec-Mag
   Label  7: Inc-Mag
   Label  8: Polygon
   Label  9: UNASSIGNED
   Label 10: UNASSIGNED
Top of body label # (1-10, 0 or // to quit)?1
Bottom of body label # (1-10, 0 or // to quit)?2
Volume magnetic susceptibility label # (1-10, 0 or // to quit)?4
Remanent or total volume magnetization label # (1-10, 0 or // to quit)?5
Declination of remanent or total magnetization label # (1-10, 0 or //
   to quit)?6
Inclination of remanent or total magnetization label # (1-10, 0 or //
   to quit)?7
You have made the following assignments:
   Height of top of body is labeled:  Top
   Height of bottom of body is labeled:  Bottom
   Volume magnetic susceptibility is labeled:  Vol-Mag
   Remanent or total volume magnetization is labeled:  Rem-Mag
   Declination of remanent or total magnetization is labeled:  Dec-Mag
   Inclination of remanent or total magnetization is labeled:  Inc-Mag
Are these assignments correct (y/n/q) [Y]?_
Name to call command file to run MAGPOLY?TEST.MAG
Output file mode (p/s/h/q) [H]?
```

II.D.2 - STANDARD GRID:

Example:

```
Output file mode (p/s/h/q) [H]?s
You have the option of writing the set of parameters for the polygons
   onto a preexisting standard grid.
Do you wish to use a preexisting grid (y/n/q)?n
Model grid?TEST.GRD
Id (up to 56 characters)?Test of POLYGON 12/18/84
Dval for grid [0.17014117E+39]?0.0
   Note: The dval is assigned to all grid intersections not interior to
         any polygons.
For outputting the standard grid you must select the label number of a
   set of parameters to use when creating the grid.
The parameter labels are:
   Label  1:  Top
   Label  2:  Bottom
   Label  3:  Density
   Label  4:  Vol-Mag
   Label  5:  Rem-Mag
   Label  6:  Dec-Mag
   Label  7:  Inc-Mag
   Label  8:  Polygon
   Label  9:  UNASSIGNED
   Label 10:  UNASSIGNED
Which parameter to use (1-10, 0 or // to quit)?3
Output file mode (p/s/h/q) [H]?
```

II.E. READ POLYGON MODEL

To read in a POLYGON model file from an earlier POLYGON session, enter the character "r" for "read" at the POLYGON command level, POLYGON responds with the following question:

Q: "Model name [ ]?"
R: Enter the file name of the POLYGON model to be read. The model will be drawn on the screen if the grid has already been drawn.

Example:

```
POLYGON command (p/o/r/w/z/h/q) [h]?_
POLYGON options available:
     p = POLYGON add/change_parm/delete/edit_poly mode
     o = Output Plouff/Godson model or standard grid
     r = Read POLYGON model from file
     w = Write POLYGON model to output file
     z = Zoom to a subgrid of grid
     q = Quit
POLYGON command (p/o/r/w/z/h/q) [p]?R
Model name []? TEST.MOD
POLYGON command (p/o/r/w/z/h/q) [p]?
```

II.F. WRITE POLYGON MODEL

To write the current information in the POLYGON model file enter the character "w" for "write" at the POLYGON command level, POLYGON responds with the following question:

Q: "Model name [ ]?"
R: Enter the name to call the POLYGON model file. If a default answer appears in brackets this name may be entered by pressing the return key.

Example:

```
POLYGON command (p/o/r/w/z/h/q) [p]?W
Model name [TEST.MOD]?TEST2.MOD
POLYGON command (p/o/r/w/z/h/q) [h]?
```

II.G. ZOOM TO SUBGRID

The zoom feature may be used to select a subgrid for redrawing on the screen at a finer scale. The zoom command stores up to five subgrid locations on a "stack" so that any of the subgrid locations may be recalled and used for zooming. The "unzoom" command is used to draw the grid without zooming, it has no effect on the stack.

The boundaries of a subgrid may be specified by one of three methods:

1) Mouse via graphics cursor,
2) Keypad cursor control keys,
3) Subgrid row and column numbers (ncmin,ncmax,nrmin,nrmax).

The subgrid is defined in methods (1) and (2) by positioning the graphics cursor and entering the screen locations of a pair of diagonal corners of the subgrid. Only the nearest grid intersections to the cursor are stored.

Example:

```
POLYGON command (p/o/r/w/z/h/q) [p]?z
*** Zoom mode ***
Zoom command (c/d/r/s/u/h/q) [H]?_
Zoom options available:
     c = Clear zoom stack
     d = Draw grid using a selected zoom value
     r = Recall and draw grid using zoom values at top of stack
     s = Select a subgrid (using mouse, cursor, or coords)
     u = Unzoom and draw grid on screen (does nothing to
         current zoom values)
     q = Quit and return to POLYGON command
Zoom command (c/d/r/s/u/h/q) [Q]?
```

II.H. QUITTING THE PROGRAM

To quit the program enter "q" for "quit" or "//" when at the POLYGON command level, POLYGON will then prompt to determine whether the current POLYGON model should be saved.

Example:

```
POLYGON command (p/o/r/w/z/h/q) [p]?q
Save model (y/n) [Y]?_
Model name [TEST.MOD]?test2.mod
$
```

NOTES ON IMPLEMENTATION

The following notes are provided to assist in converting POLYGON to a non VAX-11 FORTRAN environment or to operate on a color graphics terminal other than the Envision.

The FORTRAN code for POLYGON makes extensive use of the nonstandard DO WHILE and END DO commands of the DEC, VAX-11 FORTRAN extensions to the FORTRAN 77 standard. The DO WHILE and END DO commands may be replaced by using the IF THEN, END IF, and GO TO commands, the following example illustrates a possible conversion.

```
      DO 10 WHILE (ITEST.EQ.0)          10   IF (ITEST.EQ.0) THEN
         ...                                      ...
                                                GO TO 10
10    END DO                                 END IF
```

The following is a list of subroutines which are called within POLYGON to perform graphics operations on the Envision terminal.

ALPHON	CLOSUP	CPCHARS	CPCHARSET	CUROFF
CURON	DEFCLR	DISPLA	DRWBOX	DRWLIN
DRWPLN	DRWPLY	DRWPNT	ENTMOU	ENTPLY
ENVBCL	ENVCLR	ESCCOM	FNDCLR	GETLIN
GETMOU	GETPLY	GRAPHOFF	GTPNT	INKJET
LINCUR	LINMOU	LOADMOU	LOCCUR	MOUEXM
MOULIN	MOUSOP	MOVCUR	NCBP	RETPNT
SETCLR	SETDLM	SETFIL	SETKAM	SETLIN
SETMOU	SOFTKY			

The subroutine CLRSGD (Color standard grid) is a modified version of the GCOLOR (Grid color) program by Robert W. Simpson [unpublished]. CLRSGD draws a color contour map of a standard grid on the Envision graphics screen. The modifications enable POLYGON and CLRSGD to pass parameters via several FORTRAN COMMON blocks. The parameters are used for setting screen dimensions, grid specifications, and color and zooming parameters.

The code for CLRSGD retains GCOLOR's internal buffering technique for sending multiple commands to the Envision terminal. The subroutines listed below are called in CLRSGD to drive graphics operations on the Envision, the subroutines marked with an asterisk (*) are called only within CLRSGD.

*BINSETUP2	*BOX	DRWBOX	ENVCLR	FNDCLR
*INKJET	*RECT	*SENDCMD	SETCLR	SETFIL
NCBP				

The size of the graphics screen on the Envision terminal is ~4150x3120 pixels. POLYGON limits the display of graphics information to a slightly smaller area of ~4000x3000 pixels. The subroutine INTSCR initializes, among other things, the useable portion of the total graphics plane using the following real arrays:

```
XSCRN(1) = 0.0       Left boundary (in pixels)
XSCRN(2) = 4000.0    Right    "
```

```
YSCRN(1) = 0.0        Bottom  "
YSCRN(2) = 3000.0     TOP     "
```

POLYGON further limits graphics to a window within the above boundaries; the lower left corner is initialized by (and to):

```
XINIT = 100.0 (in pixels)
YINIT = 100.0
```

The upper right corner is defined by XSCRN(2),YSCRN(2).

Within the POLYGON program only the screen coordinates of the polygons are stored, the conversion to or from the coordinate system of the grid occurs when writing or reading a POLYGON model file.

Subroutines in POLYGON beginning with the prefix "INT" are used to initialize parameters and arrays used within the program. Each of the INT* subroutines contains a brief comment on the variables being initialized.

The subroutines GRVMOD, MAGMOD, MODOUT, and STDOUT are used to construct the output files in POLYGON. The zoom function in POLYGON is implemented completely within the FORTRAN code and is independent of the zoom feature on the Envision terminal.

Additional information about the program may be obtained by contacting the author:

Bruce A. Chuchel
USGS, MS 989
345 Middlefield Road
Menlo Park, CA, 94025
Telephone # 415-323-8111, ext. 4173.

REFERENCES

Blakely, R. J., 1981, A program for computing the magnetic anomaly over digital topography: U.S. Geological Survey Open-File Report 81-298, 46 p.

Chuchel, B A., 1985, POLYGON - Program listing: U.S. Geological Survey Open-File Report 85-233-B, 179 p.

Envision Technology, INC., 1983a, Mouse accessory reference manual, #21133-0: San Jose, CA, 8 p

Envision Technology, INC , 1983b, Models 220 and 230 color graphics terminals reference manual #20640-1: San Jose, CA, 178 p.

Godson, R. H., 1983a, MAGPOLY: A modification of DONALD PLOUFF's 3-D magnetic modelling program: U.S. Geological Survey Open-File Report 83-345, 48p.

Godson, R. H. 1983b, GRAVPOLY: A modification of DONALD PLOUFF's 3-D gravity modelling program: U.S. Geological Survey Open-File Report 83-346, 53p.

Knuth, D. E., 1969, The art of computer programming, Volume 1/Fundamental Algorithms: Menlo Park, CA, Addison-Wesley, 634p.

Parker, R. L., 1972, The rapid calculation of potential anomalies: Geophysical Journal of the Royal Astronomical Society, v. 31, p. 447-455.

Plouff, Donald, 1975a, Derivation of formulas and FORTRAN programs to compute magnetic anomalies of prisms: U.S. Geological Survey Report, 112 p., available from National Technical Information Service, No. PB-243-525, U.S. Department of Commerce, Springfield, VA, 22161.

Plouff, Donald, 1975b, Derivation of formulas and FORTRAN programs to compute gravity anomalies of prisms: U.S. Geological Survey Report, 90 p., available from National Technical Information Service, No. PB-243-526, U.S. Department of Commerce, Springfield, VA 22161.

APPENDIX A - POLYGON MODEL FILE Format

The POLYGON model file contains the following information (with the indicated formats):

```
NTOTAL     (x,i3)  - NTOTAL, the total number of polygons in the model.
LABELS  1-5     (5(x,a15))  -  The descriptive character labels associated
LABELS  6-10    (5(x,a15))     with the parameter list.
```

Next, for each polygon, the model file contains:

(i) NPOLY - An integer number used to identify the polygon.

(ii) NBRPTS - The number of vertices for this polygon.

(iii) PARM's 1-10 - The ten parameters associated with this polygon.

(iv) X(1),Y(1) - X(NBRPTS),Y(NBRPTS) - The corner-points, one X,Y pair per record; the corner points are in the coordinate system of the grid.

```
NPOLY,NBRPTS    (2(x,i3))
PARM 1-5       (5(x,e15.8))
PARM 6-10      (5(x,e15.8))
X(1),Y(1)       (2(x,e16.8))
  ...
X(NBRPTS),Y(NBRPTS)
```

Next the pointer information that is used to store the topology of the collection of polygons in a model. Appendix C describes the variables used.

```
INFO(1),IUPPER(1),IDOWN(1),ILEFT(1),IRIGHT(1)     (5(x,i3))
...
INFO(NMAX),IUPPER(NMAX),IDOWN(NMAX),ILEFT(NMAX),IRIGHT(NMAX)  -  NMAX is
      the maximum number of polygons allowed in the program.
```

The example below is an annotated listing of a POLYGON model file for the three polygons in Figure 2.

Example:

```
  3                    - Number of polygons in the model (NTOTAL).
 Top                 Bottom              Density          Vol-Mag          Rem-Mag
 Dec-Mag             Inc-Mag             Polygon          UNASSIGNED       UNASSIGNED
                          - Above is the label information for the parameters.
   1   5                  - Polygon number (NPOLY) and number of corner points
                              for polygon C (NBRPTS).
0.10000000E+01  0.00000000E+00  0.26500001E+01  0.99999997E-05  0.99999997E-04
0.16500000E+02  0.89900002E+02  0.10000000E+01  0.00000000E+00  0.00000000E+00
 0.42489571E+02    0.37232121E+02    - Corner points for polygon B, parameter
 0.42411137E+02    0.42249474E+02          information is above.
 0.50476612E+02    0.44797348E+02
 0.56162968E+02    0.36892403E+02
 2   5                  - Polygon number (NPOLY) and number of corner points
                              for polygon A (NBRPTS).
0.15000000E+01  0.60000002E+02  0.25000000E+01  0.20000001E-01  0.89000000E+02
0.14000000E+02  0.80000000E+02  0.30000000E+01  0.00000000E+00  0.00000000E+00
 0.45809875E+02    0.38669384E+02    - Corner points for polygon A, parameter
 0.45639938E+02    0.41478580E+02          information is above.
 0.48097492E+02    0.43516880E+02
```

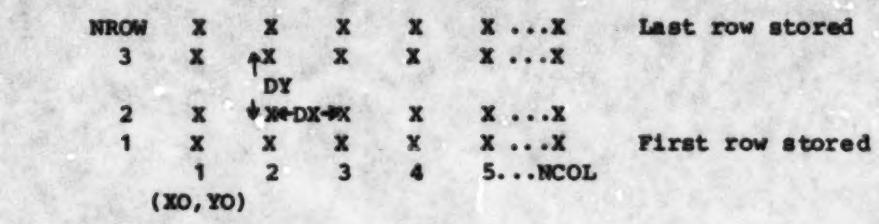
```
0.54725040E+02   0.39779995E+02
0.53365540E+02   0.37989948E+02
3   4                    - Polygon number (NPOLY) and number of corner points
                           for polygon B (NBRPTS).
0.75000000E+00  0.25000000E+00  0.26900001E+01  0.99999997E-05  0.15000001E-03
0.17000000E+02  0.89900002E+02  0.20000000E+01  0.00000000E+00  0.00000000E+00
 0.55234848E+02   0.41217258E+02      - Corner points for polygon B, parameter
 0.56254471E+02   0.44026451E+02          information is above
 0.51417805E+02   0.47083904E+02
 0.49548496E+02   0.44954140E+02
1   0   3   0   0            - Pointers for polygon C.
2   1   0   3   0            - Pointers for polygon A.
3   1   0   0   2            - Pointers for polygon B.
0   0   0   0   0                 .
0   0   0   0   0                 .
    ...                           .
0   0   0   0   0            - To NMAX.
```

APPENDIX B - Standard Grid Format

A standard grid is a two-dimensional array of binary values in the x- and y-directions. Godson [1983a] discusses how this format may be used to store more irregular two-dimensional arrays.

The file consists of two basic parts: (1) a header record and for irregular arrays, a following record that contains the x-coordinates for each column, and (2) a series of data records, each containing the column values for one row.

The following diagram shows the relationship of the grid elements in the usual case where dx and dy are positive.



A. Header record
   - ID: 56 ASCII characters of general identification.
   - PGM: 8 ASCII characters identifying the program creating the grid.
   - NCOL: Number of columns of data (integer).
   - NROW: Number of rows of data (integer).
   - NZ: Number of words per data element (integer). For single precision use 1.
   - XO: Position coordinate of first column of data (real).
   - DX: Distance between columns (real).
   - YO: Position coordinate of first row (real).
   - DY: Distance between rows (real).

B. Data record. Each data record contains one row of real data items. The total record length is NCOL times (NZ plus 1) words. The first value in the record is a dummy value usually equal to 0.0. Again, the row coordinates should be in monotonic sequence, if specified.

The subroutines REDGRD (read grid) and WRTGRD (write grid) are used in POLYGON for grid input and output.

APPENDIX C - POLYGON Data Structure

Figures 2 and 3 illustrate several simple three-body models, although much more complex models, including polygons embedded at several levels, can be envisioned. A data structure possessing the following properties was developed to provide for both the simple and complex situations.

(i) Flexible; able to change with the evolving geometry among the polygons
(ii) Memory efficient; uses the computer memory efficiently and sparingly
(iii) Search efficient; provide a framework in which a polygon can be quickly located and the algorithms which test for polygon crossing can quickly reject or emplace a polygon in the structure.

POLYGON uses a quadruply linked tree [Knuth, 1969, p. 352] to meet the above requirements. A linked tree is a collection of pointers to other objects (pointers, polygons, arrays, etc.) in the tree. POLYGON uses five integer arrays to point to other polygons in the tree; in the following discussion I, J, K, L, M are positions in the tree, n is an array indice.

INFO(n) = I - A number used to identify the polygon and which marks the position in the arrays containing the corner-point coordinates of polygon I. Note: in what follows we will assume that n=I, although this may not always be the case.

IUPPER(I) = J - Position in the tree, one level above polygon I, of the polygon that contains polygon I.

IDOWN(I) = K - Position in the tree, one level below polygon I, of the first polygon contained in polygon I.

ILEFT(I) = L - Position in the tree, at the same level as polygon I, of the polygon to the left of polygon I which is also contained in polygon J.

IRIGHT(I) = M - Position in the tree, at the same level as polygon I, of the polygon to the right of polygon I which is also contained in polygon J.

A portion of the right hand side of Figure 5 depicts the relationship among the pointers discussed above. Figure 6 shows a model which would have the data structure depicted in Figure 5.

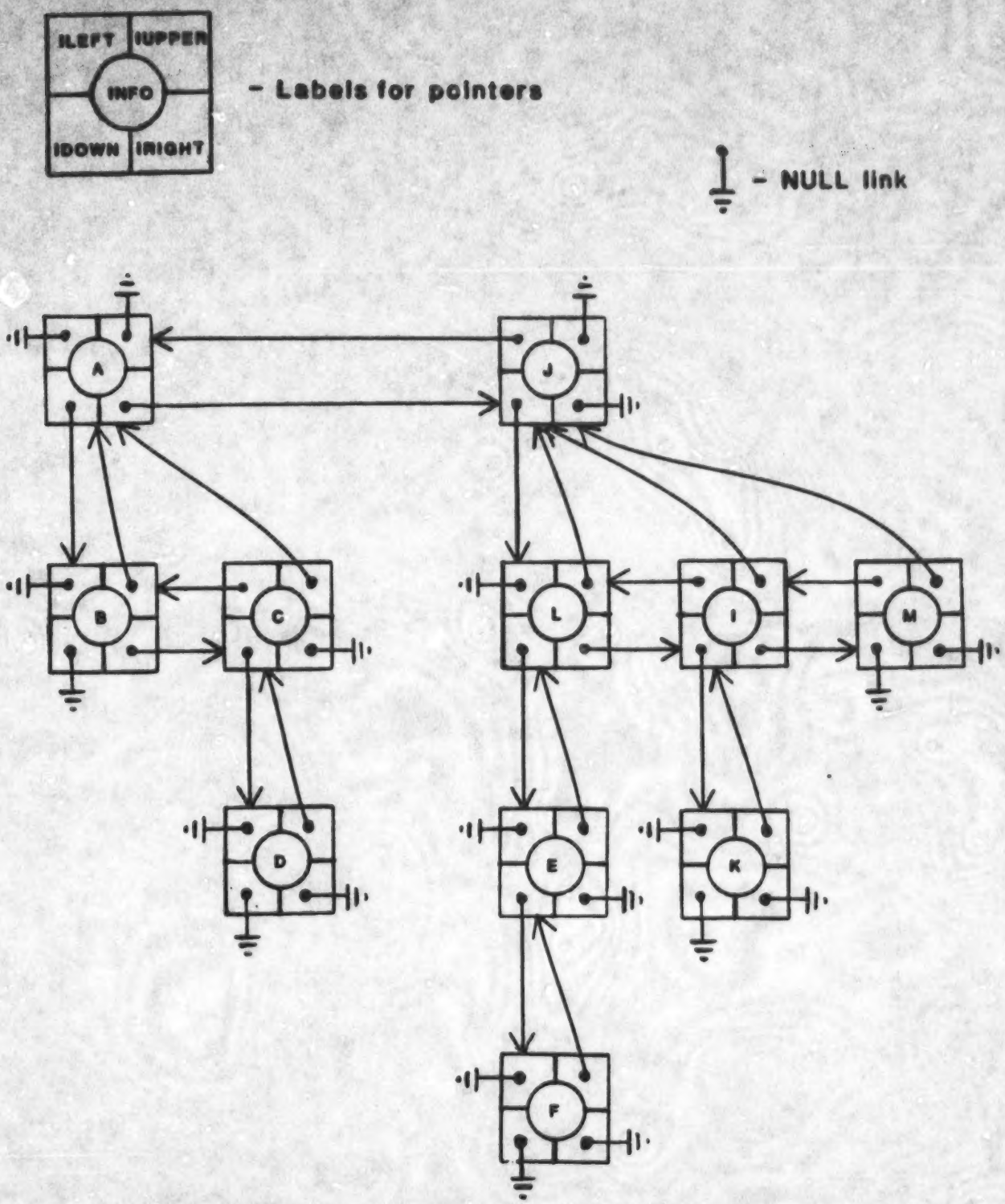


Figure 5 - Quadruply linked tree data structure for the model in Fig. 6. Knuth 1969, p. 352.

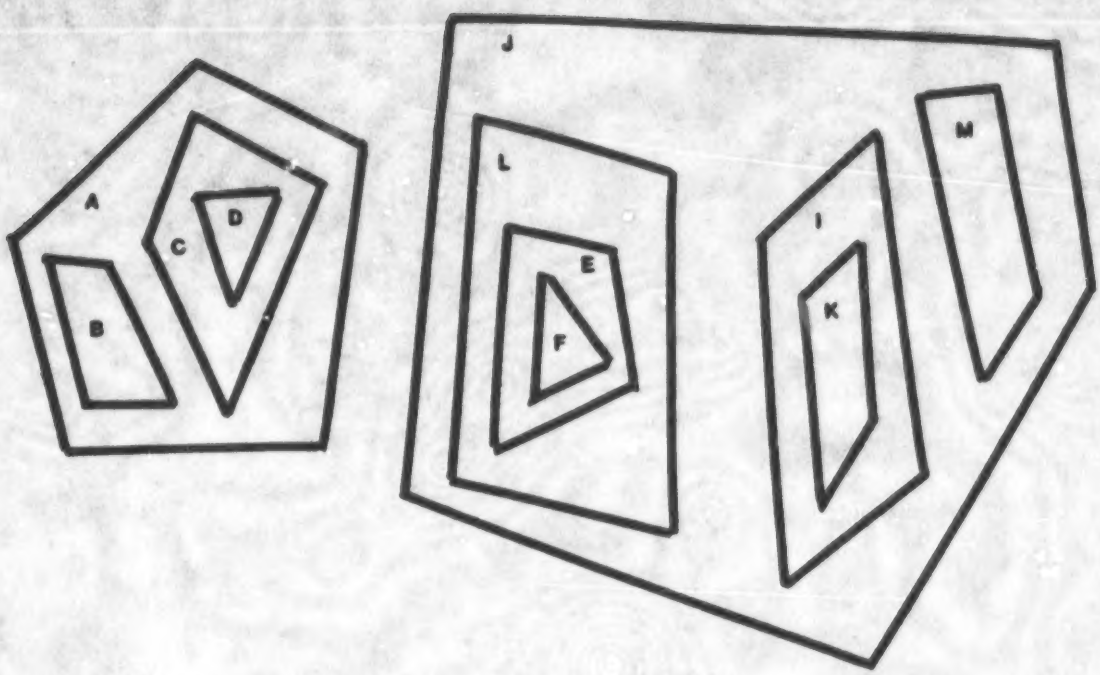


Figure 6 – A model which would have the data structure depicted in Fig. 5.